Abstract

Grouping and recognition are important components of visual scene understanding, e.g., for object detection and semantic segmentation. With end-to-end deep learning systems, grouping of image regions usually happens implicitly via top-down supervision from pixel-level recognition labels. Instead, in this paper, we propose to bring back the grouping mechanism into deep networks, which allows semantic segments to emerge automatically with only text supervision. We propose a hierarchical Grouping Vision Transformer (GroupViT), which goes beyond the regular grid structure representation and learns to group image regions into progressively larger arbitrary-shaped segments. We train GroupViT jointly with a text encoder on a large-scale image-text dataset via contrastive losses. With only text supervision and without any pixel-level annotations, GroupViT learns to group together semantic regions and successfully transfers to the task of semantic segmentation in a zero-shot manner, i.e., without any further fine-tuning. It achieves a zero-shot accuracy of 52.3% mIoU on the PASCAL VOC 2012 and 22.4% mIoU on PASCAL Context datasets, and performs competitively to state-of-the-art transfer-learning methods requiring greater levels of supervision. We open-source our code at https://github.com/NVlabs/GroupViT.

1. Introduction

Visual scenes are naturally composed of semantically-related groups of pixels. The relationship between grouping and recognition has been studied extensively in visual understanding even before the deep learning era [52,53]. In bottom-up grouping, the idea is to first re-organize pixels into candidate groups and then to process each group with a recognition module. This pipeline has been successfully applied in image segmentation from superpixels [60], constructing region proposals for object detection [75, 94] and semantic segmentation [3]. Beyond bottom-up inference, top-down feedback from recognition can also provide signals to perform better visual grouping [74, 93].

*Jiarui Xu was an intern at NVIDIA during the project.
transferring to downstream tasks [59]. The learned model can not only be transferred to ImageNet classification in a zero-shot manner and achieve state-of-the-art performance, but can also perform recognition on object categories beyond ImageNet. Inspired by this line of research, we ask the question: Can we also learn a semantic segmentation model purely with text supervision, and without any per-pixel annotations, capable of generalizing to different sets of objects categories, or vocabularies, in a zero-shot manner?

To accomplish this, we propose to bring back the grouping mechanism into deep networks, which allows semantic segments to emerge automatically with only text supervision. An overview of our approach is illustrated in Fig. 1. By training on large-scale paired image-text data with contrastive losses, we enable the model to be zero-shot transferred to several semantic segmentation vocabularies, without requiring any further annotation or fine-tuning. Our key idea is to leverage the Vision Transformer (ViT) [22] and incorporate a new visual grouping module into it.

We call our model GroupViT (Grouping Vision Transformer). Compared to convolutional neural networks (ConvNets), which operate on regular grids, the global self-attention mechanism of Transformers naturally provides the flexibility to combine visual tokens into non-grid-like segments. Instead of organizing visual tokens into grids, as recent ViT-based applications [16, 23, 44, 81] do, we propose to perform hierarchical grouping of visual tokens into irregular-shaped segments. Specifically, our GroupViT model is organized in different stages through a hierarchy of Transformer layers, where each stage contains multiple Transformers to perform information propagation among the group segments, and a grouping module that merges smaller segments into larger ones. With different input images, our model dynamically forms different visual segments, each intuitively representing a semantic concept.

We train GroupViT with text supervision only. To perform learning, we merge visual segment outputs in the final stage of GroupViT using average pooling. We then compare this image-level embedding to those derived from textual sentences via contrastive learning. We construct positive training pairs by using corresponding image and text pairs, and negative ones by using text from other images. We extract the text embedding with a Transformer model, pairs, and negative ones by using text from other images. We extract the text embedding with a Transformer model, trained jointly along with GroupViT from scratch. Interestingly, even though we only provide textual training supervision at the image level, we find that semantically meaningful segments automatically emerge using our grouping architecture.

During inference, for the task of semantic segmentation, given an input image, we extract its visual groups using GroupViT (Fig. 1). Each final group’s output represents a segment of the image. Given a vocabulary of label names for segmentation, we use the text Transformer to extract each label’s textual embedding. To perform semantic segmentation, we then assign the category labels to image segments according to their mutual similarity in the embedding space. In our experiments, we show that GroupViT trained on the Conceptual Caption [10, 63] and Yahoo Flickr Creative Commons [69] datasets with text supervision alone, can transfer to semantic segmentation tasks on the PASCAL VOC [24] and PASCAL Context [54] datasets in a zero-shot manner. Without any fine-tuning, we achieve a mean intersection over union (mIoU) of 52.3% on PASCAL VOC 2012 and an mIoU of 22.4% on PASCAL Context, performing competitively to state-of-the-art transfer-learning methods requiring greater levels of supervision. To our knowledge, our work is the first to perform semantic segmentation on different label vocabularies in a zero-shot manner with text supervision alone, without requiring any pixel-wise labels.

Our contributions are the following:

- Moving beyond regular-shaped image grids in deep networks, we introduce a novel GroupViT architecture to perform hierarchical bottom-up grouping of visual concepts into irregular-shaped groups.
- Without any pixel-level labels and training and with only image-level text supervision using contrastive losses, GroupViT successfully learns to group image regions together and transfers to several semantic segmentation vocabularies in a zero-shot manner.
- To our knowledge, ours is the first work to explore zero-shot transfer from text supervision alone to several semantic segmentation tasks without using any pixel-wise labels and establishes a strong baseline for this new task.

2. Related Work

Vision Transformer. Inspired by the success of Transformers in NLP [20, 76], the Vision Transformer (ViT) [22] was recently proposed and has been successfully applied to multiple computer vision tasks, including image classification [44, 72, 73, 86], object detection [44, 79, 88], semantic segmentation [44, 82, 91] and action recognition [4, 5, 25, 45, 61]. However, much like ConvNets, most variants of ViT still operate on regular image grids. For example, Liu et al. [44] divide the input into regular shaped windows and apply a Transformer block to each one. The convolutional operations are also inserted back into the Transformer block in [16, 23, 81]. While these variants of ViT achieve remarkable performance, they don’t fully leverage the flexibility of the global self-attention mechanism in Transformers. That is, self-attention, by design, can be applied to any arbitrary image segments and is not limited to rectangular-shaped and scan-ordered ones only. Our GroupViT model, on the other hand, leverages this property of Transformers
to learn to group visual information into several arbitrary-shaped segments. With a hierarchical design, it further merges smaller segments into larger ones and yields different semantic groups for each image.

**Representation Learning with Text Supervision.** With large-scale image-text paired data available on the Internet, representation learning with text supervision [14, 19, 30, 32, 37, 39, 49, 59, 89] has been shown to be successful in transferring to various downstream tasks such as visual question answering [2, 92] and visual reasoning [87]. For example, Desai et al. [19] pre-train ConvNets with the image captioning task, and transfer the representation by fine-tuning with downstream task annotations, e.g., object detection labels. Recently, Radford et al. [80] propose to perform contrastive learning between image and text. They show that the learned model can be directly transferred to ImageNet classification [18] in a zero-shot manner without any fine-tuning. Going beyond image classification, our GroupViT model further explores zero-shot transfer to semantic segmentation tasks with only text supervision, which has not been shown in previous work to the best of our knowledge.

**Visual Grounding.** Visual grounding aims to learn image region-text correspondence. One line of research explores a fully supervised approach to detecting text-related bounding boxes within an image [14, 27, 33, 49, 57] using datasets such as Flickr30k Entities [58] and Visual Genome [35]. To scale up learning, weakly-supervised visual grounding has been introduced where the bounding box and text correspondence is not available during training [11, 28, 41, 42, 78, 85]. However, to localize object bounding boxes these approaches still rely on pre-trained object detectors [78, 85], which, in turn, utilize box annotations from other datasets. While related, we emphasize there are two main differences between our problem setting and that of visual grounding: (i) We train our model on millions of noisy image-text pairs from the web, while visual grounding requires human curated and annotated data at a relatively smaller scale; (ii) Our GroupViT provides a bottom-up mechanism for progressive visual grouping where object segments automatically emerge with text supervision, while visual grounding needs bounding box annotations borrowed from other datasets.

**Semantic Segmentation with Less Supervision.** Multiple research directions have been proposed to learn to segment with less supervision than dense per-pixel labels. For example, few-shot learning [21, 43, 50, 55, 70, 77, 84] and active learning [8, 62, 66, 67, 83] are proposed to perform segmentation with as few pixel-wise labels as possible. Going further, zero-shot approaches [6, 38] are proposed to learn segmentation models for unseen categories without using pixel-wise labels for them. However, it still requires learning with segmentation labels on seen categories as the initial step. Another line of related research is of weakly-supervised semantic segmentation [1, 9, 26, 31, 36, 40, 65, 68, 80], which aims to learn semantic segmentation with only image-level object category supervision. While it largely reduces supervision, it still requires manual labeling using a finite vocabulary on a carefully-curated image dataset. Different from all previous work, our approach completely gets rid of human annotations and GroupViT is trained with large-scale noisy text supervision. Instead of a fixed vocabulary, we show that GroupViT can be generalized to any set of categories in a zero-shot manner for semantic segmentation.

### 3. Method

We propose the GroupViT architecture for zero-shot transfer to semantic segmentation with text supervision only. GroupViT introduces a new hierarchical grouping Transformer architecture that exploits the global self-attention mechanism of Transformers to partition input images into progressively larger arbitrary-shaped groups. We first describe GroupViT’s architecture in detail in Sec. 3.1. To train it, we employ carefully-designed contrastive losses between image-text pairs, which we describe in Sec. 3.2. Lastly, we transfer the trained GroupViT model, without further fine-tuning, to the task of zero-shot semantic segmentation as described in Sec. 3.3.

#### 3.1. Grouping Vision Transformer

We introduce the GroupViT image encoder (Fig. 2), which performs hierarchical progressive grouping of visual concepts via a Transformer-based architecture. In GroupViT, we separate Transformer layers into multiple grouping stages. In each stage, we learn a number of group tokens (as learnable parameters) via self-attention that aggregate information globally from all image tokens (segments). We then use the learned group tokens to merge similar image tokens together via a Grouping Block. Through a hierarchy of grouping stages, we group smaller image segments into larger ones. We describe each component next.

**Architecture** Following the design of ViT [22], we first split an input image into \( N \) non-overlapping patches and linearly project each into a latent space. We treat each projected patch as an input image token and denote the set of all of them as \( \{ p_i \}_{i=1}^{N} \). In each grouping stage, besides the image tokens, we concatenate a set of learnable group tokens and input them into the Transformer for that stage.

**Multi-stage Grouping** As Fig. 2(a) shows, instead of forwarding all the \( N \) input image tokens through all the layers of the Transformer, we separate its layers into a hierarchy of grouping stages. Each stage incorporates a Grouping Block at its end to merge the smaller groups into larger ones.

Formally, suppose there are \( L \) grouping stages, each indexed by \( l \) and with a set of learnable group tokens \( \{ g_i \}_{i=1}^{M_l} \).
For simplicity, we treat the image patches \( \{p_i\}_{i=1}^N \) input to the first grouping stage as the set of starting segments \( \{s_i^1\}_{i=1}^{M_0} \), where \( N = M_0 \). We simplify \( \{s_i^1\}_{i=1}^{M_{l-1}} \) to \( \{s_i^1\} \) and similarly \( \{g_i^l\}_{i=1}^{M_l} \) to \( \{g_i^l\} \). Starting with \( l=1 \), for each grouping stage, we first concatenate \( \{s_i^1\} \) and \( \{g_i^l\} \) together and then input them into a number of Transformer layers, each of which performs information propagation between them via

\[
\{\hat{g}_i^l, \hat{s}_i^l\} = \text{Transformer}(\{[\{\hat{g}_i^l\}; \{\hat{s}_i^l\}]\}),
\]

where \([ : ]\) denotes the concatenation operator. Then we group the updated \( M_{l-1} \) image segment tokens \( \{\hat{s}_i^l\} \) into \( M_l \) new segment tokens \( \{\hat{s}_i^{l+1}\}_{i=1}^{M_l} \) via a Grouping Block as

\[
\{\hat{s}_i^{l+1}\} = \text{GroupingBlock}(\{\hat{g}_i^l\}, \{\hat{s}_i^l\}).
\]

In each grouping stage \( M_l < M_{l-1} \), i.e., there are progressively fewer group tokens, resulting in progressively larger and fewer image segments. After the final grouping stage, \( L \), we apply Transformer layers on all segment tokens and finally average their outputs to obtain the final global image representation \( z^f \) as

\[
\{\hat{s}_i^{L+1}\} = \text{Transformer}(\{\hat{s}_i^{L+1}\}),
\]

\[
z^f = \text{MLP} (\text{AvgPool}(\{\hat{s}_i^{L+1}\})).
\]

As shown in Fig. 2(a), GroupViT re-organizes visual information into arbitrary image segments after the first stage itself and thus is not confined to a regular-grid structure.

**Grouping Block** As shown in Fig. 2(b), the Grouping Block at the end of each grouping stage takes the learned group tokens and image segment tokens as inputs. It merges all the segment tokens that are assigned to the same group token into a single new image segment, based on similarity in the embedding space.

Formally, we compute the similarity matrix \( A^l \) between the group tokens \( \{\hat{g}_i^l\} \) and segment tokens \( \{\hat{s}_i^l\} \) via a Gumbel-Softmax \([29, 51]\) operation computed over the group tokens as

\[
A^l_{ij} = \frac{\exp(W_g\hat{g}_i^l W_k \hat{s}_j^l + \gamma_i)}{\sum_{k=1}^{M_l} \exp(W_g\hat{g}_i^l W_k \hat{s}_j^l + \gamma_k)},
\]

where \( W_g \) and \( W_k \) are the weights of the learned linear projections for the group and segment tokens, respectively, and \( \{\gamma_i\} \) are i.i.d random samples drawn from the Gumbel \((0, 1)\) distribution. We compute the group to assign a segment token to by taking the one-hot operation of \( \text{argmax} \) over all the groups. Since the one-hot assignment operation via \( \text{argmax} \) is not differentiable, we instead use the straight through trick in \([56]\) to compute the assignment matrix as

\[
\hat{A}^l = \text{one-hot}(A^l_{\text{argmax}}) + A^l - \text{sg}(A^l),
\]

where \( \text{sg} \) is the stop gradient operator. With the straight through trick, \( \hat{A}^l \) has the one-hot value of assignment to a single group, but its gradient is equal to the gradient of \( A^l \), which makes the Grouping Block differentiable and end-to-end trainable. We call this one-hot assignment strategy.
as hard assignment. After assigning the segment tokens to the different learned groups, we merge the embedding of all the tokens belonging to the same group to form a new segment token $s_i^{l+1}$. For each group, the output of the Grouping Block is a weighted sum of the segment tokens assigned to that group and computed as

$$s_i^{l+1} = g_i^l + W_o \sum_{j=1}^{M_l} A_{i,j} W_v \theta_j^l,$$

where $W_o$ and $W_v$ are the learned weights to project the merged features. An alternative to hard assignment is soft assignment, which uses $A^l$ instead of $\tilde{A}^l$ for computing Eqn. 5. Empirically, we found that hard assignment results in more effective grouping versus soft assignment (Table 1).

The Grouping Block works similarly to a single iteration of the previously proposed Slot Attention mechanism [46]. While Slot Attention learns instance-level grouping from self-supervision, our Grouping Block groups similar semantic regions with weak text supervision. For example, in the second row of Fig. 6, the two horses are grouped together.

### 3.2. Learning from Image-Text Pairs

To train GroupViT to perform hierarchical grouping, we employ carefully-designed contrastive losses between image-text pairs. We describe these next.

**Image-Text Contrastive Loss** To learn visual representations via text supervision, following [30, 59], we train a dual-encoder architecture via an image-text contrastive loss. In our case, GroupViT acts as the image encoder and a Transformer [76] as the text encoder. The final image embedding from GroupViT (Eqn. 2) is the average embedding of all its output segment tokens. The text embedding is the embedding of the last output token (end-of-sentence token) from the text Transformer. We forward the input image and text in a pair through their respective encoders, project them into a common embedding space and compute a similarity measure between them. We consider all matched image-text pairs as positive pairs, and all other unmatched ones as negative ones. Our training objective is to pull the representations of the positive pairs closer to each other, while pushing those of the unmatched ones far away from each other.

Formally, assume a batch of $B$ image-text pairs $\{(x_i^l, x_i^T)\}_{i=1}^B$, where $x_i^l$ and $x_i^T$ are the image and text inputs, respectively, of the $i$-th pair. We encode each of them, via their respective encoders, into embedding vectors $z_i^l$ and $z_i^T$ and $l_2$-normalize each. We then measure their similarity by computing their dot product. The total image-text contrastive loss is defined as

$$L_{I \rightarrow T} = L_{I \rightarrow T} + L_{T \rightarrow I},$$

which is composed of an image-to-text contrastive loss defined as

$$L_{I \rightarrow T} = -\frac{1}{B} \sum_{i=1}^{B} \log \frac{\exp(z_i^l \cdot z_i^T / \tau)}{\sum_{j=1}^{B} \exp(z_i^l \cdot z_j^T / \tau)},$$

and a text-to-image contrastive loss defined as

$$L_{T \rightarrow I} = -\frac{1}{B} \sum_{i=1}^{B} \log \frac{\exp(z_i^T \cdot z_i^l / \tau)}{\sum_{j=1}^{B} \exp(z_j^T \cdot z_i^l / \tau)},$$

where $\tau$ is a learnable temperature parameter to scale the logits.

**Multi-Label Image-Text Contrastive Loss** To enable effective visual grouping, besides the image-text loss in Eqn. 6, we propose a multi-label contrastive loss with text prompting. As illustrated in Fig. 3, we use the “prompting engineering” mechanism proposed in [59] to generate additional text labels for each image besides its originally provided sentence label. Specifically, we randomly select $K \ln$ words from a sentence $x_i^T$, and prompt each of them with a set of handcrafted sentence templates, e.g., “A photo of a {noun}”. The motivation to select nouns is that objects in images are more likely to be described by them. Besides training with the original image-text pairs $\{(x_i^l, x_i^T)\}_{i=1}^B$, we employ additional contrastive losses between the new sets of image-“prompted text” pairs $\{(x_i^l, x_i^{T_1})\}_{i=1}^B, \{(x_i^l, x_i^{T_2})\}_{i=1}^B, \ldots, \{(x_i^l, x_i^{T_K})\}_{i=1}^B$, where $x_i^{T_k}$ are all prompted sentences generated from the nouns sampled from $x_i^T$. As shown in Fig. 3, compared to the standard contrastive loss (Eqn. 6), which results in only one positive pair among the batch $B$, in our case, each image $x_i^l$ has $K$ positive text pairs and $K(B - 1)$ negative ones.
Similarly to the standard image-text contrastive loss (Eqn. 6), our multi-label contrastive loss is defined as
\[
\mathcal{L}_{I \rightarrow \{T_k\}^K_{k=1}} = \mathcal{L}_{I \rightarrow \{T_k\}^K_{k=1}} + \mathcal{L}_{\{T_k\}^K_{k=1} \rightarrow I},
\]
(7)
which is a sum of two two-way contrastive losses
\[
\mathcal{L}_{I \rightarrow \{T_k\}^K_{k=1}} = -\frac{1}{B} \sum_{i=1}^{B} \sum_{k=1}^{K} \log \frac{\exp(z_i^T \cdot T_k / \tau)}{\sum_{j=1}^{B} \exp(z_i^T \cdot z_j^T / \tau)}
\]
and
\[
\mathcal{L}_{\{T_k\}^K_{k=1} \rightarrow I} = -\frac{1}{K \cdot B} \sum_{k=1}^{K} \sum_{i=1}^{B} \log \frac{\exp(T_k \cdot z_i^T / \tau)}{\sum_{j=1}^{B} \exp(z_i^T \cdot z_j^T / \tau)}.
\]
Finally, the total image-text contrastive loss for training GroupViT is defined as
\[
\mathcal{L} = \mathcal{L}_{I \rightarrow T} + \mathcal{L}_{I \rightarrow \{T_k\}^K_{k=1}}.
\]
(8)

3.3. Zero-Shot Transfer to Semantic Segmentation

Since GroupViT automatically groups images into semantically-similar segments, its output can be easily zero-shot transferred to semantic segmentation without any further fine-tuning. This zero-shot transfer pipeline is illustrated in Fig. 4. To infer the segments of an image belonging to a finite vocabulary of object classes, we forward a test image through GroupViT without applying AvgPool to its final L output segments, and obtain the embedding of each of them as \( \{z_i^T\}_i \). Each segment token correspond to an arbitrarily-shaped region of the input image. We then compute the similarity between the embedding of each segment token and the text embedding of all the semantic classes present in the dataset. We assign each image segment to the semantic class with the highest image-text similarity.

Specifically, let \( \hat{A}^l \) be the assignment matrix of the \( l \)th grouping stage described in Sec. 3.1, which indicates the mapping between the input and output segments of \( l \)th stage. Multiplying all the stage-level assignment matrices \( \prod_{l=1}^{L} \hat{A}^l \) yields the final assignment between the input patches \( \{x_i\}_i \) and the final-stage output tokens \( \{z_i^T\}_i \). We use the same “prompting engineering” as described in Sec. 3.2 to transform all the semantic segmentation label names into sentences. The embedding of label names in the dataset is \( \{z_i^C\}_i \), where \( C \) is the number of classes. As shown in Fig. 4, to classify an image segment \( z_i^T \) to one of \( C \) classes, we compute the dot product between \( i_2 \)-normalized class name embedding vectors \( \{z_i^C\}_i \) and \( z_i^T \), and assign it to the class with the highest similarity.

4. Experiments

4.1. Implementation Details

Architecture The architecture of GroupViT is based on ViT-S [22, 72] with 12 Transformer layers, each with a hidden dimension of 384. We use input images of size 224 × 224 and a patch size of 16 × 16. We add a learnable positional embedding to each patch after linearly projecting it. We experiment with 1-stage and 2-stage architectures for GroupViT. Both architectures output 8 tokens after the final grouping stage. In 1-stage GroupViT, we learn 64 group tokens and insert the grouping block after the sixth Transformer layer. Before the grouping block, we project the 64 group tokens into 8 tokens using an MLP-Mixer layer [71] and output 8 segment tokens. In 2-stage GroupViT, there are 64 and 8 group tokens in the first and second grouping stages, respectively. We insert grouping blocks after the sixth and ninth Transformer layers. Our text-encoder is the same as [59]. We use a 2-layer MLP to project the visual and text embedding vectors into the same latent space.

Training We use the CC [10, 63] and the filtered YFCC [69] datasets for training, containing 12M and 14M image-text pairs, respectively. Our batch size is 4096 with a learning rate initialized to 0.0016 and decayed via the cosine schedule [48]. We use the Adam [34] optimizer with a weight decay of 0.05. We train GroupViT for 30 epochs with the 5 initial epochs containing linear warm-up. For the multi-label contrastive loss, we set \( K = 3 \). We use the same text templates as in [59] for generating prompts.

Zero-shot Transfer to Semantic Segmentation We evaluate GroupViT for the task of zero-shot transfer to semantic segmentation on the validation splits of the PASCAL VOC 2012 [24] and PASCAL Context [54] datasets. They each contain 20 and 59 foreground classes, respectively, with an additional background class. During inference, GroupViT predicts only the foreground classes by thresholding the softmax-normalized-similarity between the embedding of the output image segments and the text segmentation labels, where we set the threshold to 0.95 and 0.35 for PASCAL VOC 2012 and PASCAL Context, respectively. We resize each input image to have a shorter side length of 448.
### Table 1. Ablation results of hard vs. soft assignment and multi-label contrastive loss.

<table>
<thead>
<tr>
<th>arch</th>
<th>hard assignment</th>
<th>multi-label loss</th>
<th>mask mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupViT</td>
<td>✔</td>
<td>✔</td>
<td>12.0</td>
</tr>
<tr>
<td>GroupViT</td>
<td>✔</td>
<td>✔</td>
<td>36.7</td>
</tr>
<tr>
<td>GroupViT</td>
<td>✔</td>
<td>✔</td>
<td>25.1</td>
</tr>
<tr>
<td>GroupViT</td>
<td>✔</td>
<td>✔</td>
<td>39.3</td>
</tr>
</tbody>
</table>

### Table 2. Ablation results of different numbers of group and output tokens.

<table>
<thead>
<tr>
<th>arch</th>
<th># group tokens</th>
<th># output tokens</th>
<th>mask mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupViT</td>
<td>16</td>
<td>4</td>
<td>28.6</td>
</tr>
<tr>
<td>GroupViT</td>
<td>16</td>
<td>8</td>
<td>37.1</td>
</tr>
<tr>
<td>GroupViT</td>
<td>32</td>
<td>8</td>
<td>38.3</td>
</tr>
<tr>
<td>GroupViT</td>
<td>64</td>
<td>8</td>
<td>39.3</td>
</tr>
<tr>
<td>GroupViT</td>
<td>64</td>
<td>16</td>
<td>38.0</td>
</tr>
</tbody>
</table>

### Table 3. Ablation results of single-stage and multi-stage grouping.

<table>
<thead>
<tr>
<th>arch</th>
<th># stages</th>
<th>mask mIoU</th>
<th>boundary mIoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupViT</td>
<td>1</td>
<td>39.3</td>
<td>31.6</td>
</tr>
<tr>
<td>GroupViT</td>
<td>2</td>
<td>41.1</td>
<td>33.5</td>
</tr>
<tr>
<td>GroupViT</td>
<td>1</td>
<td>37.2</td>
<td>32.3</td>
</tr>
<tr>
<td>GroupViT</td>
<td>2</td>
<td>52.3</td>
<td>40.3</td>
</tr>
</tbody>
</table>

### 4.2. Ablation Study

To discern the contribution of each component of GroupViT, we conduct an ablation study. For all experiments, we train a 1-stage GoupViT with the CC dataset, unless otherwise specified. We report mIoU (mean intersection over union) of the predicted and ground truth segmentation masks on the PASCAL VOC 2012 validation set.

**Hard vs. Soft Assignment** In each Grouping Block, we assign image segment tokens to group tokens via hard or soft assignment (Sec. 3.1). For soft assignment, we use the original $A^l$ matrix instead of $\hat{A}^l$ used for hard assignment to compute Eqn. 5. The impact of this is shown in the first column of Table 1. We find that hard assignment improves over soft assignment by a large margin, $>10\%$ mIoU. We conjecture that with soft assignment, the features of new segment tokens $\{s_i^{l+1}\}$ are likely to be more correlated with each other due to absence of zero values in $A^l$. Hence, each group may contain information from the same image patches increasing ambiguity while assigning text labels to image segments. With hard assignment, however, the affinity matrix $\hat{A}^l$ assigns image segments to groups in a mutually exclusive manner, making groups more differentiated and their assignment to text labels less ambiguous.

**Multi Label Contrastive Loss** We investigate the effect of adding the multi-label contrastive loss in the second column of Table 1. Adding the multi-label contrastive loss to the standard one (Eqn. 8) improves performance both with hard and soft assignment, by 13.1% and 2.6%, respectively. With the multi-label contrastive loss, the input text during training and inference is in a similar prompted format. We conjecture that this consistency helps GroupViT better classify the learned image segments into label categories.

**Group Tokens** In Table 2, we compare different group and output tokens. We observe that increasing group tokens consistently improves performance. Conceptually, each group token represents a distinct semantic concept. So more group tokens presumably help GroupViT learn to group more semantic concepts. Note that although the number of group tokens is much less than the number of classes in the real world, each group token is a feature vector in a 384-D embedding space, which can represent many more concepts than just 1. We also experiment with different output tokens and find 8 to be optimal, similar to findings in [61].

**Multi Stage Grouping** In Table 3, we compare the 1-stage and 2-stage GroupViT architectures. We also compare their visual zero-shot semantic segmentation results in Fig. 5. We find that the 2-stage GroupViT generates smoother segmentation maps compared to its 1-stage counterpart. To quantify the smoothness of the segmentation maps, we also report the boundary mIoU [15] in Table 3, which computes the IoU of boundaries only. The 2-stage GroupViT improves the mask mIoU of the 1-stage variant by 1.8% and the boundary mIoU by 1.9%. We also train both models on a combination of the CC [10] and YFCC [69] datasets.
While the 1-stage model does not benefit as much from the expanded dataset, the 2-stage model improves significantly both in terms of the mask and boundary mIoU values by $\sim$7%. These results demonstrate that our hierarchical grouping mechanism is effective, especially when training with larger datasets. We adopt the 2-stage GroupViT in the following experiments.

### 4.3. Comparisons with Existing Methods

We compare the zero-shot semantic segmentation performance of GroupViT with other zero-shot baselines and with methods for fully supervised transfer, based on ViT-S.

**Comparison with Zero-Shot Baselines** We train ViT and a text encoder with the image-text contrastive loss defined in CLIP [59], for comparison. To zero-shot transfer CLIP to semantic segmentation, during inference, we first apply non-parametric grouping on its output features. We then compute the similarity between the average feature of each group and the text embedding of the segmentation labels of the dataset. In this way, any non-parametric grouping method for ViT combined with CLIP can be considered as a zero-shot semantic segmentation baseline. We also report a “pixel-wise” baseline, which treats each pixel as a group and performs classification independently. As Table 4 shows that GroupViT outperforms other grouping methods by a large margin. This demonstrates that, compared to ViT trained with CLIP, our GroupViT is more effective at zero-shot transfer to semantic segmentation. In the supplement, we also show that GroupViT’s ImageNet classification performance is comparable to that of ViT.

**Comparison with Fully-Supervised Transfer** We compare the performance of GroupViT with fully-supervised transfer to semantic segmentation. For fully-supervised transfer, we fine-tune a semantic segmentation head jointly with a pre-trained representation [12,90] on the training sets of the PASCAL VOC 2012 and PASCAL Context datasets separately and report their performances in Table 5. For a fair comparison, we employ a ViT architecture comparable to GroupViT’s for all baselines. Specifically, we append a $1 \times 1$ convolution layer to a pre-trained ViT, trained with images of size $224 \times 224$ and fine-tune the whole network with ground truth masks for 4k iterations. During inference, we resize the input images to have a shorter side length of 448 pixels. For fully-supervised transfer, we compare both supervised and self-supervised pre-training methods against GroupViT (Table 5). GroupViT (without fine-tuning) outperforms all variants of ViT pre-trained with self-supervision (with supervised fine-tuning) by a large margin on PASCAL VOC 2012 and is comparable to them on PASCAL Context. This implies that GroupViT, without any pixel-level annotations is able to transfer to several semantic segmentation datasets and can outperform existing state-of-the-art transfer-learning methods requiring more supervision (i.e., pixel-level labels for supervised transfer). Interestingly, on PASCAL VOC 2012, the zero-shot performance of GroupViT (mIoU of 52.3%) approaches that of fully-supervised ViT (mIoU of 53%) trained with both image classification and segmentation labels, which is significant. We show selected qualitatively segmentation results of GroupViT in Fig. 6 and more in the supplement.

### 5. Conclusion

We take the first step towards learning semantic segmentation with text alone and without any explicit human supervision. We show that, with GroupViT, the representation learned from large-scale noisy image-text pairs can be transferred to semantic segmentation in a zero-shot manner. This work also demonstrates that besides image classification, text supervision could also be transferred to finer-grained vision tasks, which hasn’t yet been explored previously and opens up an exciting research direction.

**Acknowledgements.** Prof. Wang’s lab is supported, in part, by grants from NSF CCF-2112665 (TILOS).